



The image shows a grid of musical notes arranged in a pattern. The notes are represented by various symbols: 'MM' (Mezzo-Mezzo), 'AA' (Alto-Alto), 'CC' (Canto-Canto), 'LL' (Basso-Basso), and 'SS' (Soprano-Soprano). The grid consists of several rows and columns of these symbols, creating a visual representation of musical harmony and structure.

(2)	76	DECLARATIONS
(3)	116	MCALL
(4)	151	MAC\$IMPLMCALL
(5)	266	MAC\$GET_MLB_LIN
(6)	293	LIBRY
(7)	377	MAC\$SYSLIB
(8)	425	ALLOCATE MCF BLOCK

EXPLICIT MACRO CALL ROUTINE  
IMPLICIT MACRO CALL ROUTINE  
READ LINE FROM MACRO LIBRARY  
PROCESS .LIBRARY DIRECTIVE  
SET UP THE SYSTEM LIBRARY

0000 1 .TITLE MAC\$MACLIB MACRO LIBRARY PROCESSOR  
0000 2 .IDENT 'V04-000'  
0000 3 .  
0000 4 .  
0000 5 .\*\*\*\*\*  
0000 6 .\*: \*  
0000 7 .\*: COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0000 8 .\*: DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0000 9 .\*: ALL RIGHTS RESERVED.  
0000 10 .\*: \*  
0000 11 .\*: THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0000 12 .\*: ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0000 13 .\*: INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0000 14 .\*: COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0000 15 .\*: OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0000 16 .\*: TRANSFERRED.  
0000 17 .\*: \*  
0000 18 .\*: THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0000 19 .\*: AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0000 20 .\*: CORPORATION.  
0000 21 .\*: \*  
0000 22 .\*: DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0000 23 .\*: SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0000 24 .\*: \*  
0000 25 .\*: \*  
0000 26 .\*\*\*\*\*  
0000 27 .\*: \*  
0000 28 .\*: \*  
0000 29 .\*: \*\*  
0000 30 .\*: FACILITY: VAX MACRO ASSEMBLER OBJECT LIBRARY  
0000 31 .\*: \*  
0000 32 .\*: ABSTRACT:  
0000 33 .\*: \*  
0000 34 .\*: The VAX-11 MACRO assembler translates MACRO-32 source code into object  
0000 35 .\*: modules for input to the VAX-11 LINKER.  
0000 36 .\*: \*  
0000 37 .\*: ENVIRONMENT: USER MODE  
0000 38 .\*: \*  
0000 39 .\*: AUTHOR: Benn Schreiber, CREATION DATE: 24-AUG-78  
0000 40 .\*: \*  
0000 41 .\*: MODIFIED BY:  
0000 42 .\*: \*  
0000 43 .\*: V03.01 MTR0016 Mike Rhodes 07-Jun-1982  
0000 44 .\*: Add logic to MAC\$IMPLMCALL to save/modify/restore the  
0000 45 .\*: listing directive for listing macro definitions. This  
0000 46 .\*: corrects a problem where the .NOSHOW MD directive caused  
0000 47 .\*: the listing to lose the invocation lines or other assorted  
0000 48 .\*: parts when there were no macros explicitly defined in the  
0000 49 .\*: program text and the macros were being defined from a library.  
0000 50 .\*: Also included are fixes for broken branch destinations.  
0000 51 .\*: \*  
0000 52 .\*: V02.17 MTR0001 Mike Rhodes 09-Nov-1981  
0000 53 .\*: Add default name descriptor use in opening the system  
0000 54 .\*: library STARLET.  
0000 55 .\*: \*  
0000 56 .\*: V02.16 PCG0007 P. George 20-Aug-1981  
0000 57 .\*: Report .LIBRARY errors in listing and with exit status.

0000	58	:	
0000	59	:	V01.15 RN0023 R. Newland 3-Nov-1979
0000	60	:	New message codes to get error messages from system
0000	61	:	message file.
0000	62	:	
0000	63	:	V01.14 RN0011 R. Newland 11-Sep-1979
0000	64	:	New librarian support
0000	65	:	
0000	66	:	V01.11 004 B. Schreiber 10-JAN-1979
0000	67	:	Correct "illegal block size" error if macro library
0000	68	:	has MNT entries allocated but none used.
0000	69	:	
0000	70	:	V01.12 008 B. Schreiber 23-JAN-1979
0000	71	:	Clear index number of pages for SYSLIB FDB on setup.
0000	72	:	
0000	73	:	V01.13 RN0003 R. Newland 18-Feb-1979
0000	74	--	Correct initial values of macro library binary
			chop routine

```
0000 76 .SBTTL DECLARATIONS
0000 77 ; INCLUDE FILES:
0000 78 ;
0000 79 ;
0000 80 ;
0000 81 ;
0000 82 ;
0000 83 ;
0000 84 ;
0000 85     $MAC_CTLFLGDEF      ; DEFINE CONTROL FLAGS
0000 86     $MAC_GRAMMARDEF    ; DEFINE TERMINAL GRAMMAR SYMBOLS
0000 87     $MAC_MLFDEF        ; Define MLF offsets
0177 88     $MAC_GENVALDEF    ; DEFINE GENERAL VALUES
0177 89     $MAC_INTCODDEF    ; DEFINE INT. FILE CODES
0177 90     $MAC_SYMBOLKDEF   ; DEFINE SYMBOL BLOCK OFFSETS
0000 91     $NAMDEF           ; DEFINE NAM OFFSETS
0000 92     $RABDEF           ; DEFINE RAB OFFSETS
0000 93     $RMEDEF           ; DEFINE RME
0000 94     $FABDEF           ; DEFINE FAB OFFSETS
0000 95     $MACMSGDEF       ; Define message codes
0000 96 ;
0000 97 ;
0000 98 : LOCAL DATA
0000 99 ;
0000 100 ;
00000000 101     .PSECT MAC$RO_DATA,NOEXE,NOWRT,GBL,LONG
0000 102 ;
0000 103     MLF_ARGLIST:
0000 104         $ARGLST 2,MLF_SIZE,MAC$GL_BASEADDR
000C 105 ;
0000177 106     MLF_SIZE:
000C 107         .LONG MLFSK_BLKSIZ
0010 108 ;
00000000 109     .PSECT MAC$RW_DATA,NOEXE,LONG
0000 110 ;
00000001'00000000 111     MAC$TMPSYMDS:          ; Temp sym descriptor (used for
0000 112         .LONG 0,MAC$AB_TMPSYM+1      ; LBR$LOOKUP_KEY calls)
0008 113 ;
00000000 114     .PSECT MAC$RO_CODE_MAC,NOWRT,GBL,LONG
```

	0000	116	.SBTTL MCALL	EXPLICIT MACRO CALL ROUTINE	
	0000	117			
	0000	118	:++		
	0000	119	: FUNCTIONAL DESCRIPTION:		
	0000	120	:		
	0000	121	: THIS ROUTINE IS CALLED WHEN THE MCALL DIRECTIVE IS FOUND.		
	0000	122	: ANY MACRO NAMES FOLLOWING THE DIRECTIVE THAT ARE NOT DEFINED		
	0000	123	: ARE DEFINED BY DOING IMPLICIT MACRO CALLS ON THEM.		
	0000	124	:		
	0000	125	:--		
	0000	126			
	0000	127	MCALL::	DIRECTIVE = KMCALL	
50	0000'CF	01	C3	SET UP ERROR POINTER	
	0000'CF	50	D0		
	FFF2'	30	000B		
	08 50	E8	000E		
	FFE7'	31	0016		
53	0000'CF	9E	0019	10\$:	SCAN A MACRO NAME
	FFDF'	30	001E	134	BRANCH IF WE FOUND ONE
	OF 50	E8	0021	135	Else directive syntax error
	21	10	0024	136	ISSUE ERROR AND RETURN
OD	58	D1	0026	137	LOOK UP NAME IN MACRO HASH TABLE
	08	13	0029	138	IF IT IS THERE WE CAN SAVE SOME TIME
	FFCD'	30	0030	139	BRANCH IF FOUND
	FFCA'	30	0033	140	NO--DO AN IMPLICIT MACRO CALL
2C	5A	91	0036	141	DID WE FIND IT?
	06	12	0039	142	IF EQL YES
	FFC2'	30	003B	143	No--set message code
	FFBF'	30	003E	144	ISSUE MESSAGE TO PASS 2
OD	5A	91	0041	145	SKIP SPACES
	BA	12	0044	146	STOP ON A COMMA?
	05	0046	147	30\$:	IF NEQ NO
			148		YES--SKIP IT
			149		THEN SKIP SPACES
					GET TO EOL?
					IF NEQ NO--KEEP GOING
					YES--MCALL IS DONE

MA  
PS  
--  
-  
SA  
MA  
MA  
MAPS  
--  
-  
SA  
MA  
MA  
MAPh  
--  
In  
Co  
Pa  
Sy  
Pa  
Sy  
Ps  
Cr  
As  
Th  
55  
Th  
45  
17Ma  
--  
-S  
TO  
11  
Th  
MA

0047 151 .SBTTL MAC\$IMPLMCALL IMPLICIT MACRO CALL ROUTINE  
 0047 152  
 0047 153 :++  
 0047 154 : FUNCTIONAL DESCRIPTION:  
 0047 155 :  
 0047 156 : THIS ROUTINE IS CALLED EITHER BY 'MCALL' OR BY THE MAC\$SYMBOL  
 0047 157 : ROUTINE WHEN AN UNIDENTIFIED NAME IS DETECTED. THE IN-CORE  
 0047 158 : INDICES OF ALL 'KNOWN' MACRO LIBRARIES ARE SEARCHED STARTING  
 0047 159 : WITH THE LAST ONE SPECIFIED. THE FOLLOWING METHOD IS USED:  
 0047 160 :  
 0047 161 : 1) THE IN-CORE INDEX OF EACH MLB IS SEARCHED USING  
 0047 162 : A MATCHC INSTRUCTION. IF THE NAME IS FOUND, THE  
 0047 163 : MLB IS OPENED AND THE MACRO IS DEFINED.  
 0047 164 :  
 0047 165 : 2) IF THE NAME IS NOT FOUND IN ANY OF THE MACRO  
 0047 166 : LIBRARIES, A TOKEN OF 'ERR03' IS RETURNED.  
 0047 167 :  
 0047 168 :--  
 0047 169 :  
 0047 170 MAC\$IMPLMCALL::  
 56 0000'CF 57 DD 0047 171 PUSHL R7 :SAVE R7  
 0000'CF 66 9E 0049 172 MOVAB W^MAC\$AB\_TMPSYM,R6 :POINT TO THE NAME WE ARE LOOKING FOR  
 57 0000'CF 9E 004E 173 MOVZBW (R6),W^MAC\$TMPSYMD\$ :Set up descriptor for macro name  
 0053 174 MOVAB W^MAC\$GL\_MLB\_QUE,R7 :POINT TO MLB QUE HEADER  
 0058 175 :  
 0058 176 NEXT\_MLB:  
 0058 177 MOVL (R7),R7 :LINK TO NEXT MLB FDB  
 0058 178 CMPL R7,#MAC\$GL\_MLB\_QUE :ARE WE DONE?  
 07 12 0062 179 BNEQ 10\$ :IF NEQ NO  
 57 8ED0 0064 180 POPL R7 :YES--RESTORE R7  
 58 03 DO 0067 181 MOVL #ERR03,R8 :RETURN ERROR TOKEN  
 006A 182 RSB :  
 006B 183 :  
 006B 184 : Call librarian procedure to look up macro name  
 006B 185 :  
 006B 186 10\$:  
 0000'CF 9F 006B 187 PUSHAB W^MAC\$GL\_TXTRFA :Address to store text RFA  
 0000'CF 9F 006F 188 PUSHAB W^MAC\$TMPSYMD\$ :Address of name descriptor  
 14 A7 9F 0073 189 PUSHAB MLFSL CTINDEX(R7) :Address of control table index  
 00000000'GF 03 FB 0076 190 CALLS #3,G^BRSLOOKUP\_KEY :Look-up macro name  
 D8 50 E9 007D 191 BLBC R0,NEXT\_MLB :Not found if LBC  
 0000'CF 57 DO 0080 192 MOVL R7,W^MAC\$GL\_MLFPTR :Save MLF pointer  
 0085 193 :  
 FF78' 30 0085 194 BSBW MAC\$ALL\_1\_PAGE :ALLOCATE A VIRTUAL PAGE  
 50 DD 0088 195 PUSHL R0 :SAVE ITS ADDRESS  
 60 0000'CF 0086 8F 28 008A 196 MOVC3 #LSTSK\_BUFSIZ,W^MAC\$AB\_LINEBF,(R0) :COPY CURRENT LINE OUT  
 83 0000'CF DO 0092 197 MOVL W^MAC\$GL\_LINENUM,(R3)+ :SAVE CURRENT STATE  
 83 FFFFFFF7 8F CB 0097 198 BICL3 #^C<FLGSM\_CONT>,(R11),(R3)+ :SAVE CONTINUATION STATE  
 83 5A DO 009F 199 MOVL R10,(R3)+ :SAVE CURRENT CHARACTER  
 83 0000'CF DO 00A2 200 MOVL W^MAC\$GL\_LINEPT,(R3)+ :SAVE LINE POINTER  
 83 0000'CF DO 00A7 201 MOVL W^MAC\$GL\_LINLEN,(R3)+ :SAVE LINE LENGTH  
 FF51' 30 00AC 202 BSBW MAC\$ALL\_T\_PAGE :ALLOCATE AN INPUT BLOCK  
 60 0000'CF DO 00AF 203 MOVL W^MAC\$GE\_INPUTP,(R0) :LINK TO LAST INPUT BLOCK  
 0000'CF 80 DE 00B4 204 MOVAL (R0)+,W^MAC\$GL\_INPUTP :MAKE NEW BLOCK CURRENT BLOCK  
 80 80 D4 00B9 205 CLRL (R0)+ :CLEAR TEXT LINK  
 80 017F'CF 9E 00BB 206 MOVAB W^MAC\$GET\_MLB\_LIN,(R0)+ :SET NEW LINE ROUTINE  
 00C0 207 :

00C0 208 ; THE REST OF THE INPUT BLOCK IS NOT USED. HENCE IT IS NOT INITIALIZED.

00C0 209 ;

5A 0D 9A 00C0 210 MOVZBL #CR,R10 :FORCE READING OF NEW LINE  
FF3A' 30 00C3 211 MAC\$GETCHR :GET FIRST CHARACTER OF MACRO  
FF37' 30 00C6 212 MACSSYMSCNUP :SCAN FIRST SYMBOL  
03 50 E8 00C9 213 BLBS R0 20S :Branch if symbol was found  
00A5 31 00CC 214 BRW MAC LIB FMT ERR :ELSE If not found then error  
FF27' 30 00D6 215 20\$: MOVAB MAC\$AL\_PRMH\$HTB,R3 :LOOK IN PERMANENT SYMBOL TABLE  
03 50 E8 00D9 216 BSBW MACSSRC\$YMTAB  
0095 31 00DC 217 BLBS R0 30S :Branch if symbol was found  
50 8F 0B A1 91 00DF 218 BRW MAC LIB FMT ERR :ELSE, If not found then error  
03 13 00E4 219 30\$: CMPB SYMSB\_TOKEN[R1],#KMACRO :WAS it ".MACRO" ?  
008B 31 00E6 220 BEQL 40S : If EQL yes.  
00000005'GF DD 00E9 221 40\$: PUSHL G^LSTSG\_MACRODEF+SYMSL\_VAL :SAVE THE LISTING STATE,  
0005'CF FF 8F 98 00EF 222 223 CVTBL #-1,W^LSTSG\_MACRODEF+SYMSL\_VAL :ENABLE LISTING MACRO DEF's.  
00F5 224 SINTOUT\_LW INT\$ SETLONG,- :INDICATE THE (POSSIBLE) SWITCH...  
FEF6' 30 0107 225 <G^LSTSG\_MACRODEF+SYMSL\_VAL, #LSTSG\_MACRODEF+SYMSL\_VAL> ;...TO PASS  
010A 226 BSBW MACRO : Define the MACRO  
010A 227 :  
010A 228 : NOW RESTORE THINGS TO PRE-MLB READING STATE  
010A 229 :  
00000005'GF 8ED0 010A 230 POPL G^LSTSG\_MACRODEF+SYMSL\_VAL :RETRIEVE THE PRIOR LISTING STATE  
0111 231 SINTOUT\_LW INT\$ SETLONG,- :INDICATE THE (POSSIBLE) SWITCH...  
0111 232 <G^LSTSG\_MACRODEF+SYMSL\_VAL, #LSTSG\_MACRODEF+SYMSL\_VAL> ;...TO PASS  
0000'CF D6 0123 233 INCL W^MACSGL\_MLB\_MDF :COUNT MACRO DEFINED FROM MACRO LIBRARY  
08 A7 D6 0127 234 INCL MLFSL MCDEF(R7) :ALSO COUNT IN MLB FDB  
50 0000'CF D0 012A 235 MOVL W^MACSGL\_INPUTP, R0 :POINT TO THE INPUT BLOCK  
0000'CF 60 D0 012F 236 MOVL (R0), W^MACSGL\_INPUTP :RESTORE LAST INPUT BLOCK  
FEC9' 30 0134 237 BSBW MACSDEA\_1\_PAGE :DEALLOCATE INPUT BLOCK  
50 6E D0 0137 238 MOVL (SP), R0 :GET ADDRESS OF SAVED LINE BLOCK  
0000'CF 60 0086 8F 28 013A 239 MOVC3 #LST\$K\_BUFSIZ,(R0), W^MAC\$AB\_LINEBF ;RESTORE INPUT LINE  
0000'CF 81 D0 0142 240 MOVL (R1)+, W^MACSGL\_LINENUM :RESTORE LINE NUMBER  
81 D5 0147 241 TSTL (R1)+ :CHECK CONT FLAG  
06 6B 03 E3 0148 242 BEQL 50S :IF EQL CLEAR IT  
04 11 014F 243 BBCS #FLGSV\_CONT,(R11),60S :NO--SET IT  
00 6B 03 E5 0151 245 50\$: BBCC #FLGSV\_CONT,(R11),60S :CLEAR CONT FLAG  
5A 81 D0 0155 246 60\$: MOVL (R1)+, R10 :RESTORE CURRENT CHARACTER  
0000'CF 81 D0 0158 247 MOVL (R1)+, W^MACSGL\_LINEPT :RESTORE LINE POINTER  
0000'CF 81 D0 015D 248 MOVL (R1)+, W^MACSGL\_LINELN :RESTORE LINE LENGTH  
50 8ED0 0162 249 POPL R0 :RESTORE ADDRESS OF SAVE PAGE  
57 8ED0 0165 250 POPL R7 :RESTORE R7  
58 0D DO 0168 251 MOVL #MACTXT,R8 :RETURN TOKEN FOR MACRO  
FE92' 30 016B 252 BSBW MACSDEA\_1\_PAGE :DEALLOCATE PAGE  
51 0000'CF DO 016E 253 MOVL W^MACSGE\_MACPTR,R1 :RETURN PTR TO MNB IN R1  
05 0173 254 RSB :RETURN  
0174 255 :  
0174 256 : MACRO LIBRARY FORMAT ERROR. THE CODE TO REPORT THE ERROR IN PASS 2  
0174 257 : IS EMITTED TO THE INTERMEDIATE FILE, AND THE ASSEMBLY IS TERMINATED.  
0174 258 : PASS 2 IS THEN EXECUTED TO REPORT THE ERRORS.  
0174 259 :  
0174 260 :  
0174 261 MAC\_LIB\_FMT\_ERR: : Get the message code  
0174 262 SMAC\_ERR MACLBFMTER :REPORT ERROR TO PASS 2  
FEB4' 30 0179 263 BSBW MAC\$ERRORPT :  
FEB1' 31 017C 264 BRW MAC\$ABORT\_PASS1 :ABORT PASS 1

L 7

```

017F 266 .SBTTL MAC$GET_MLB_LIN READ LINE FROM MACRO LIBRARY
017F 267
017F 268 :++
017F 269 : FUNCTIONAL DESCRIPTION:
017F 270 :
017F 271 : THIS ROUTINE IS CALLED BY MAC$GETCHR WHEN IT IS TIME TO
017F 272 : READ ANOTHER MACRO DEFINITION LINE FROM A MACRO LIBRARY.
017F 273 : THE LINE IS PLACED IN THE INPUT BUFFER MAC$AB_LINELN.
017F 274 :
017F 275 :--
017F 276 :
017F 277 MAC$GET_MLB_LIN::
51 0000'CF D0 017F 278 MOVE W^MAC$GL_MLFPTR,R1 ; Get current MLF pointer
      0000'CF 9F 0184 279 PUSHAB W^MAC$GL_LINELN ; Address to store line length
      0000'CF 9F 0188 280 PUSHAB W^MAC$GL_LINEBFDS ; Address of buffer descriptor
      14 A1 9F 018C 281 PUSHAB MFLSL CTINDEX(R1) ; Address of control table index
00000000'GF 03 FB 018F 282 CALLS #3,G^BRSGET RECORD ; Get record
      DB 50 E9 0196 283 BLBC R0,MAC LIB FMT ERR ; IF LBC THEN LIBRARY ERROR
      0000'CF D6 0199 284 10$: INCL W^MAC$GL_M[B]_GET ; COUNT ANOTHER MLB GET
50 0000'CF D0 019D 285 MOVL W^MAC$GL_LINELN,R0 ; Get line length
      DB 13 01A2 286 BEQL MAC$GET_MLB_LIN ; SKIP NULL LINES
      51 0000'CF 9E 01A4 287 MOVAB W^MAC$AB_LINEBF,R1 ; POINT TO THE LINE BUFFER
      0000'CF 51 D0 01A9 288 MOVL R1,W^MAC$GL_LINEPT ; SET UP THE LINE POINTER
      0000'CF 51 D0 01AE 289 MOVL R1,W^MAC$GL_ERRPTX ; AND THE ERROR TOKEN POINTER
      6041 0D 90 01B3 290 MOVB #CR,(R0)[R1] ; STORE CR AT END OF LINE
      05 01B7 291 RSB

```

01B8 293 .SBTTL LIBRY  
 01B8 294  
 01B8 295 :++  
 01B8 296 : FUNCTIONAL DESCRIPTION:  
 01B8 297 :  
 01B8 298 : THIS ROUTINE IS CALLED TO PROCESS THE .LIBRARY DIRECTIVE.  
 01B8 299 : THE FILENAME WITHIN THE DELIMITERS IS SCANNED. THE FILE  
 01B8 300 : IS THEN OPENED, AND AN IN-CORE INDEX IS CREATED. THE FDB  
 01B8 301 : FOR THE NEW MLB IS THEN ADDED TO THE FRONT OF THE MLB QUEUE.  
 01B8 302 :--  
 01B8 303 :--  
 01B8 304 : LIBRY:::  
 58 DD 01B8 305 LIBRY::: DIRECTIVE = KLIBRARY  
 OD FE43' 30 01BA 306 PUSHL R8  
 SA 91 01BD 307 BSBW MAC\$SKIPSP  
 07 12 01C0 308 CMPB R10,#CR  
 22 11 01C2 309 BNEQ 10\$  
 58 56 5A 90 01C9 310 5\$: SMAC\_ERR DIRSYNX  
 0008'CF 9E 01CC 311 BRB 40\$  
 6B 01 C8 01D1 312 10\$: MOVAB R10,R6  
 01D1 313 MOVAB W^MAC\$AB\_TMPBUF+8,R8  
 01D4 314 BISL2 #FLGSM\_ALLCHR,(R11)  
 01D4 315 : Point to temp buffer to accumulate name  
 01D4 316 : (a descriptor will be formed at start)  
 01D4 317 : PASS SEMI-COLONS IN CASE SOME  
 FE29' 30 01D4 318 20\$: TURKEY PUTS A VERSION NUMBER ON  
 56 5A 91 01D7 319 BSBW MAC\$GETCHR  
 18 13 01DA 320 CMPB R10,R6  
 OD 5A 91 01DC 321 BEQL 50\$  
 05 13 01DF 322 CMPB R10,#CR  
 88 5A 90 01E1 323 BEQL 30\$  
 EE 11 01E4 324 MOVAB R10,(R8)+  
 01E4 325 BRB 20\$  
 01E6 326 : HERE IF CR BEFORE DELIMITER  
 01E6 327 :  
 6B 58 8ED0 01E6 328 30\$: SMAC\_ERR UNTERMARG : Set message code  
 01 CA 01EB 329 40\$: POPL R8 : RESTORE R8  
 FEOC' 31 01EE 330 BICL2 #FLGSM\_ALLCHR,(R11) : CLEAR ALLCHR FLAG  
 01F1 331 BRW MAC\$ERRORPT : ISSUE ERROR AND RETURN  
 01F4 332 :  
 01F4 333 : HERE WHEN DELIMITER SEEN  
 01F4 334 :  
 6B 01 CA 01F4 335 50\$: BICL2 #FLGSM\_ALLCHR,(R11) : CLEAR ALLCHR FLAG  
 FE06' 30 01F7 336 BSBW MAC\$GETCHR : SKIP OVER DELIMITER  
 FE03' 30 01FA 337 BSBW MAC\$SKIPSP : SKIP SPACES  
 OD 5A 91 01FD 338 CMPB R10,#CR : STOP ON EOL?  
 CO 12 0200 339 BNEQ 5\$ : IF NEQ NO--LOSE  
 0000'CF 58 00000008'8F C3 0202 340 SUBL3 #MAC\$AB\_TMPBUF+8,R8, - : Form descriptor to library  
 0004'CF 0008'CF 9E 020C 341 W^MAC\$AB\_TMPBUF : file name  
 00C5 30 0213 342 MOVAB W^MAC\$AB\_TMPBUF+8, -  
 0213 343 W^MAC\$AB\_TMPBUF+4  
 0213 344 BSBW MAC\$ALL\_ALB\_MLF : Get memory block for MLF  
 0216 345 :  
 0216 346 : Call librarian procedure to initialise library  
 0216 347 :  
 18 A6 9F 0216 348 PUSHAB MLFSX\_NAMBLK(R6) : Address of NAM block  
 0000'CF 9F 0219 349 PUSHAB W^MAC\$GL\_LIBTYPE : Address of type = MLB

				DIRECTIVE				
0000'CF	9F	021D	350	PUSHAB	W^MAC\$GL_LIBFUNC	; Address of function = READ		
14 A6	9F	0221	351	PUSHAB	MLFSL CTINDEX(R6)	; Address of control table index		
00000000'GF	04	FB	0224	352	CALLS	#4,G^C\$RSINI_CONTROL	; Initialise library	
22 50	E9	022B	353	BLBC	R0,55\$	; Error if LBC		
			354					
			355			; Call librarian procedure to open library file		
OC A6	9F	022E	356	PUSHAB	MLFSQ_FNAMDS(R6)	; Address of resultant length		
OC A6	9F	0231	357	PUSHAB	MLFSQ_FNAMDS(R6)	; Address of resultant descriptor		
00	DD	0234	358	PUSHL	#0	; No related file name		
0000'CF	9F	0236	359	PUSHAB	W^MAC\$MLB_DEFNAM	; Address of default name		
00	DD	023A	360	PUSHL	#0	; No create options		
0000'CF	9F	023C	361	PUSHAB	W^MAC\$AB_TMPBUF	; Address of file name		
14 A6	9F	0240	362	PUSHAB	MLFSL CTINDEX(R6)	; Address of control table index		
00000000'GF	07	FB	363	CALLS	#7,G^C\$RSOPEN	; Open library file		
58 8ED0		0243	364	POPL	R8	; RESTORE R8		
14 50	E8	024A	365	BLBS	R0,60\$	; BRANCH IF GOOD OPEN		
		0250	366					
56	DD	0250	367					
00000000'EF	01	FB	368	PUSHL	R6	; Address of MLF		
		0252	369	CALLS	#1,MAC\$ERR_LBOPEN	; Report library open error		
FD9F'	30	0259	370	SMAC_ERR	MLBOPNERR	; REPORT ERROR WITHIN LISTING		
FD9C'	31	0261	371	BSBW	MAC\$ERRORPT	; AND EXIT WITH ERROR STATUS		
0000'CF	66	0E	0264	372	BRW	MAC\$ABORT_PASS1	; GO ABORT PASS 1	
0000'CF	D6	0269	373	60\$:	INSQUE	(R6),W^MAC\$GL_MLB_QUE	; INSERT IN MLB QUEUE	
	05	026D	374		INCL	W^MAC\$GL_MLB_CNT	; Increment macro library count	
			375		RSB			
				55\$:				

026E 377 .SBTTL MAC\$SYSLIB\_SET SET UP THE SYSTEM LIBRARY

026E 378

026E 379 ++

026E 380 : FUNCTIONAL DESCRIPTION:

026E 382 : THIS ROUTINE IS CALLED AT MACRO-32 INITIALIZATION TO ENSURE  
026E 383 : THAT THE SYSTEM LIBRARY EXISTS, AND CREATES AN IN-CORE INDEX  
026E 384 : OF IT. THE MACRO LIBRARY QUEUE IS ALSO INITIALIZED.

026E 385 :--

026E 386 :--

026E 387 :--

026E 388 MAC\$SYSLIB\_SET::

50 0000'CF 9E 026E 389 MOVAB W^MAC\$GL\_MLB\_QUE,R0 ;INIT THE MLB QUEUE  
60 60 DE 0273 390 MOVAL (R0), (R0) ...  
60 80 DE 0276 391 MOVAL (R0)+ (R0) ...  
56 0000'CF 9E 0279 392 MOVAB W^MAC\$SYSLIB\_MLF\_R6 ; Point to SYSLIB MLF entry  
14 A6 D4 027E 393 CLRL MLFSL CTINDEX(R6) ; Clear control table index  
OC A6 00FF 6F B0 0281 394 MOVW #MLFSR\_RSFLN,MLFSQ\_FNAMDS(R6) ; Initialise file  
10 A6 78 A6 DE 0287 395 MOVAL MLFST\_FNAM(R6),MLFSQ\_FNAMDS+4(R6) ; name descriptor  
028C 396

028C 397 : Call librarian procedure to initialise library  
028C 398 :

18 A6 9F 028C 399 PUSHAB MLFSX\_NAMBLK(R6) ; Address of NAM block  
0000'CF 9F 028F 400 PUSHAB W^MAC\$GL\_LIBTYPE ; Address of type = MLB  
0000'CF 9F 0293 401 PUSHAB W^MAC\$GL\_LIBFUNC ; Address of function = READ  
14 A6 9F 0297 402 PUSHAB MLFSL CTINDEX(R6) ; Address of control table index  
00000000'GF 04 FB 029A 403 CALLS #4,G^BRSINI\_CONTROL ; Initialise library  
2B 50 E9 02A1 404 BLBC R0,10\$ ; Error if LBC

02A4 405

02A4 406 : Call librarian procedure to open library file  
02A4 407 :

0C A6 9F 02A4 408 PUSHAB MLFSQ\_FNAMDS(R6) ; Address of resultant length  
0C A6 9F 02A7 409 PUSHAB MLFSQ\_FNAMDS(R6) ; Address of resultant descriptor  
00 00 00 00 00 00 00 410 PUSHL #0 ; No related file name  
00000000'GF 9F 02AA 411 PUSHAB G^MAC\$SYSLIB\_DFN ; Address of default name descriptor  
00 00 00 00 00 00 00 412 PUSHL #0 ; No create options  
0000'CF 9F 02B4 413 PUSHAB W^MAC\$SYSLIB\_FNM ; Address of file name descriptor  
14 A6 9F 02B8 414 PUSHAB MLFSL CTINDEX(R6) ; Address of control table index  
00000000'GF 07 FB 02B8 415 CALLS #7,G^BRSOPEN ; Open library file  
0A 50 E9 02C2 416 BLBC R0,10\$ ; IF LBC THEN ERROR  
0000'CF 66 0E 02C5 417 INSQUE (R6),W^MAC\$GL\_MLB\_QUE ; INSERT IN MLB QUEUE  
0000'CF 00 D6 02CA 418 INCL W^MAC\$GL\_MLB\_CNT ; Increment macro library count  
05 02CE 419 RSB

02CF 420 10\$:

00000000'EF 56 DD 02CF 421 PUSHL R6 ; Address of MLF  
01 FB 02D1 422 CALLS #1,MAC\$ERR\_LBOPEN ; Report library open error  
FD25. 31 02D8 423 BRW MAC\$LAST\_CHANCE ; and go die

02DB 425 .SBTTL ALLOCATE MLF BLOCK  
 02DB 426  
 02DB 427 Functional description:  
 02DB 428 This routine is called to allocate a block of memory for  
 02DB 429 a MLF (Macro Library File) entry. The allocated memory block  
 02DB 430 is zeroed and the NAM block and file name descriptors initialised.  
 02DB 431  
 02DB 432  
 02DB 433 Inputs:  
 02DB 434 None  
 02DB 435  
 02DB 436 Outputs:  
 02DB 437 R6 = Address of MLF  
 02DB 438  
 02DB 439  
 02DB 440  
 02DB 441 --  
 02DB 442  
 02DB 443 MAC\$ALL\_MLB\_MLF:::  
 00000000'GF 00000000'EF FA 02DB 444 CALG MLF\_ARGLIST,G^LIB\$GET\_VM ; Get memory block  
 1F 50 E9 02E6 445 BLBC R0,TOS ; Error if LBC  
 66 0177 8F 56 0000'CF DO 02E9 446 MOVL W^MAC\$GL\_BASEADDR,R6 ; Get base address of memory block  
 OC A6 00 6E 00 2C 02EE 447 MOVC \$#0,(SP),R0,#MLFSK\_BLKSIZ,(R6) ; Clear MLF  
 10 A6 0OFF 8F B0 02F6 448 MOVW #MLFSK\_RSFNLN,MLFSQ\_FNAMDS(R6) ; Initialise file name  
 6002 8F B0 0301 449 MOVAL MLFST\_FNAM(R6),MLFSQ\_FNAMDS+4(R6) ; descriptor  
 18 A6 0305 450 MOVW #<<NAME\_SC\_BLN>>8+NAMSC\_BID>,- ; Identify NAM block  
 05 0307 451 MLFSX\_NAMBLK(R6)  
 0308 452 RSB  
 0308 453 10\$: CALLS #0,W^MAC\$ERR\_NOMEM\_0 ; Report the error  
 FCF0' 00 FB 0308 454 BRW MAC\$LAST\_CHANCE ; and go die  
 0310 31 030D 455  
 0310 456  
 0310 457 .END

SST1 = 00000002  
 SCOUNT = 00000038  
 ARG\$K\_SIZE = 000003E8  
 AUD\$K\_SIZE = 00000010  
 BLNK = 00000020  
 CHR\$M\_COMMA CR = 00000020  
 CHR\$M\_ILL\_CRR = 00000040  
 CHR\$M\_NUM\_BER = 00000010  
 CHR\$M\_SPA\_MSK = 00000001  
 CHR\$M\_SYM\_CH1 = 00000008  
 CHR\$M\_SYM\_CHR = 00000004  
 CHR\$M\_SYM\_DLM = 00000002  
 CHR\$V\_COMMA CR = 00000005  
 CHR\$V\_CVTLWC = 00000061  
 CHR\$V\_ILL\_CHR = 00000006  
 CHR\$V\_NOCVT = 0000007F  
 CHR\$V\_NUM\_BER = 00000004  
 CHR\$V\_SPA\_MSK = 00000000  
 CHR\$V\_SYM\_CH1 = 00000003  
 CHR\$V\_SYM\_CHR = 00000002  
 CHR\$V\_SYM\_DLM = 00000001  
 CNT = 00000002  
 CR = 0000000D  
 DAND = 0000001D  
 DANGCLS = 00000016  
 DANGOPN = 00000015  
 DAT = 00000020  
 DBUP = 0000002B  
 DCLS = 00000018  
 DCOLON = 00000010  
 DCOMMA = 0000000F  
 DDIV = 0000001C  
 DEOL = 0000000B  
 DEQ = 00000011  
 DGUP = 0000002C  
 DINTEGER = 00000022  
 DIUP = 0000002D  
 DLUP = 0000002E  
 DMASK = 00000032  
 DMINUS = 0000001A  
 DOPCODE = 0000000E  
 DOPN = 00000017  
 DOR = 0000001E  
 DPC = 00000012  
 DPLUS = 00000019  
 DPOUND = 00000021  
 DSQCLS = 00000014  
 DSQOPN = 00000013  
 DSUP = 0000002F  
 DTIMES = 0000001B  
 DUPA = 00000023  
 DUPB = 00000024  
 DUPC = 00000025  
 DUPD = 00000026  
 DUPF = 00000028  
 DUPM = 00000029  
 DUPO = 00000027

DUPX = 0000002A  
 DWUP = 00000030  
 DXOR = 0000001F  
 ERR = 00000000  
 ERR01 = 00000001  
 ERR02 = 00000002  
 ERR03 = 00000003  
 ERR04 = 00000004  
 ERR05 = 00000005  
 ERR06 = 00000006  
 ERR07 = 00000007  
 ERR08 = 00000008  
 ERR09 = 00000009  
 FF = 0000000C  
 FLG\$M\_ALLCHR = 00000001  
 FLG\$M\_BOL = 00000002  
 FLG\$M\_CHKLPND = 00100000  
 FLG\$M\_COMPEXPR = 00000004  
 FLG\$M\_CONT = 00000008  
 FLG\$M\_CRF = 40000000  
 FLG\$M\_CRSEEN = 00000001  
 FLG\$M\_DATRPT = 00000010  
 FLG\$M\_DBGOUT = 00004000  
 FLG\$M\_DLIMSTR = 00008000  
 FLG\$M\_ENDMCH = 00000020  
 FLG\$M\_EVALEXPR = 00000040  
 FLG\$M\_EXPOPT = 00000080  
 FLG\$M\_EXTERR = 00010000  
 FLG\$M\_EXTWRN = 00020000  
 FLG\$M\_FIRSTLN = 00000200  
 FLG\$M\_IFSTAT = 00800000  
 FLG\$M\_IIF = 00400000  
 FLG\$M\_INSERT = 00000100  
 FLG\$M\_IRPC = 20000000  
 FLG\$M\_LEXOP = 00000002  
 FLG\$M\_LSTXST = 00000200  
 FLG\$M\_MAC2COL = 00000800  
 FLG\$M\_MACL = 00000800  
 FLG\$M\_MACLTB = 08000000  
 FLG\$M\_MACTXT = 00010000  
 FLG\$M\_MEBLST = 00001000  
 FLG\$M\_MOREARG = 00002000  
 FLG\$M\_MOREINP = 00000008  
 FLG\$M\_NEWPND = 00000400  
 FLG\$M\_NOREF = 01000000  
 FLG\$M\_NTYPPEPC = 00000020  
 FLG\$M\_NULCHR = 00040000  
 FLG\$M\_OBJXST = 00200000  
 FLG\$M\_OPNDCHK = 00000100  
 FLG\$M\_OPRND = 00002000  
 FLG\$M\_OPTVFLIDX = 00001000  
 FLG\$M\_ORDLST = 00020000  
 FLG\$M\_P2 = 00004000  
 FLG\$M\_RPTIRP = 10000000  
 FLG\$M\_SEQFIL = 02000000  
 FLG\$M\_SKAN = 00008000  
 FLG\$M\_SPECOP = 00000004

```

FLGSM_SPLALL      = 04000000
FLGSM_STOIMF      = 00040000
FLGSM_SYM2COL     = 00000400
FLGSM_TOCFLG      = 00080000
FLGSM_UPAFLG      = 00000010
FLGSM_UPDFIL      = 00000080
FLGSM_UPMARG      = 00000040
FLGSM_XCRF        = 80000000
FLGSV_ALLCHR      = 00000000
FLGSV_BOL         = 00000001
FLGSV_CHKLPND    = 00000014
FLGSV_COMPEXPR    = 00000002
FLGSV_CONT        = 00000003
FLGSV_CRF         = 0000001E
FLGSV_CRSEEN      = 00000020
FLGSV_DATRPT      = 00000004
FLGSV_DBGOUT      = 0000002E
FLGSV_DLIMSTR     = 0000002F
FLGSV_ENDMCH      = 00000005
FLGSV_EVAL^XPR    = 00000006
FLGSV_EXPORT       = 00000007
FLGSV_EXTERR      = 00000030
FLGSV_EXTRWN      = 00000031
FLGSV_FIRSTLN     = 00000029
FLGSV_IFSTAT       = 00000017
FLGSV_IIF          = 00000016
FLGSV_INSERT       = 00000008
FLGSV_IRPC         = 0000001D
FLGSV_LEXOP        = 00000021
FLGSV_LSTXST       = 00000009
FLGSV_MAC2COL      = 0000002B
FLGSV_MACL         = 00000008
FLGSV_MACLTB       = 0000001B
FLGSV_MACTXT       = 00000010
FLGSV_MEBLST       = 0000000C
FLGSV_MOREARG      = 0000002D
FLGSV_MOREINP      = 00000023
FLGSV_NEWPND       = 0000000A
FLGSV_NOREF        = 00000018
FLGSV_NTYPEDC      = 00000025
FLGSV_NULCHR       = 00000032
FLGSV_OBJXST       = 00000015
FLGSV_DPNDCCHK     = 00000028
FLGSV_OPRND        = 0000000D
FLGSV_OPTVFLIDX    = 0000002C
FLGSV_ORDLST       = 00000011
FLGSV_P2            = 0000000E
FLGSV_RPTIRP       = 0000001C
FLGSV_SEQFIL       = 00000019
FLGSV_SKAN          = 0000000F
FLGSV_SPECOP        = 00000022
FLGSV_SPLALL       = 0000001A
FLGSV_STOIMF       = 00000012
FLGSV_SYM2COL       = 0000002A
FLGSV_TOCFLG        = 00000013
FLGSV_UPAFLG        = 00000024
FLGSV_UPDFIL        = 00000027

```

```

FLGSV_UPMARG      = 00000026
FLGSV_XCRF         = 0000001F
GOALS?             = 0000000A
HASHSZ              = 0000007F
HYPHEN              = 0000002D
ID                  = 0000000C
INPSK_BUFSIZ       = 000003E8
INTSK_BUFSIZ       = 000013F4
INTSK_BUFWRN       = 00001390
INTS_ADD            = 00000001
INTS_AND            = 00000002
INTS_ASH            = 00000003
INTS ASN            = 0000000C
INTS AUGPC          = 0000000D
INTS BDST            = 0000000E
INTS CHKL            = 0000000F
INTS DIV             = 00000004
INTS END             = 00000010
INTS EPT             = 00000011
INTS ERR             = 00000012
INTS ETX             = 00000013
INTS FNEWL           = 00000014
INTS ILG              = 00000000
INTS INFO             = 0000003A
INTS LGLAB           = 00000015
INTS MACL             = 00000016
INTS MUL             = 00000005
INTS NEG             = 00000006
INTS NEWL           = 00000017
INTS NEWP           = 00000018
INTS NOT             = 00000007
INTS OP               = 00000019
INTS OR               = 00000008
INTS PRIL            = 0000001A
INTS PRT              = 0000001B
INTS PSECT           = 0000001C
INTS REDEF           = 0000001D
INTS REF              = 0000001E
INTS REST             = 0000001F
INTS SAME             = 00000009
INTS SAVE             = 00000020
INTS SBTTL           = 00000021
INTS SETFLAG          = 00000022
INTS SETLONG          = 00000023
INTS SPIC              = 00000024
INTS SPID             = 00000025
INTS STIB              = 00000026
INTS STIL              = 00000028
INTS STIW              = 00000027
INTS STKEPT           = 00000029
INTS STKG              = 0000002A
INTS STKL              = 0000002B
INTS STKPC             = 0000002C
INTS STKS              = 0000002D
INTS STOB              = 00000034
INTS STOL              = 0000002E
INTS STOW              = 00000035

```

INT\$\_STRB  
 INT\$\_STRL  
 INT\$\_STRSB  
 INT\$\_STRSW  
 INT\$\_STRW  
 INT\$\_STSB  
 INT\$\_STSW  
 INT\$\_SUB  
 INT\$\_SUME  
 INT\$\_WRN  
 INT\$\_XOR  
 KADDRESS  
 KALIGN  
 KASCIC  
 KASCID  
 KASCII  
 KASCIZ  
 KBLKA  
 KBLKB  
 KBLKD  
 KBLKF  
 KBLKG  
 KBLKH  
 KBLKL  
 KBLKO  
 KBLKQ  
 KBLKW  
 KBYTE  
 KCROSS  
 KDEBUG  
 KDFLT  
 KDOUBLE  
 KDSABL  
 KENABL  
 KEND  
 KENDC  
 KENDM  
 KENDR  
 KENTRY  
 KERROR  
 KEVEN  
 KEXTRN  
 KFIELD  
 KFLOAT  
 KGFLOAT  
 KGLOBL  
 KHFLOAT  
 KIDENT  
 KIF  
 KIFF  
 KIFT  
 KIFTF  
 KIIF  
 KINCLUDE  
 KIRP  
 KIRPC  
 KLIBRARY

= 0000002F  
 = 00000031  
 = 00000032  
 = 00000033  
 = 00000030  
 = 00000036  
 = 00000037  
 = 0000000A  
 = 00000039  
 = 00000038  
 = 0000000B  
 = 00000037  
 = 0000005A  
 = 00000033  
 = 00000078  
 = 00000034  
 = 00000035  
 = 0000003F  
 = 00000040  
 = 00000041  
 = 00000042  
 = 0000007E  
 = 0000007F  
 = 00000043  
 = 00000080  
 = 00000044  
 = 00000038  
 = 00000079  
 = 00000055  
 = 0000007B  
 = 00000039  
 = 00000056  
 = 00000057  
 = 00000076  
 = 0000004E  
 = 00000053  
 = 0000004F  
 = 00000058  
 = 00000071  
 = 0000005B  
 = 0000005D  
 = 0000003A  
 = 0000003B  
 = 00000081  
 = 0000005E  
 = 00000082  
 = 0000006A  
 = 00000046  
 = 00000048  
 = 00000049  
 = 0000004A  
 = 00000047  
 = 0000005F  
 = 0000004B  
 = 0000004C  
 = 00000060

KLINK  
 KLIST  
 KLONG  
 KMACRO  
 KMCALL  
 KMDELETE  
 KMEXIT  
 KNARG  
 KNCHR  
 KNCROS  
 KNLIST  
 KNTYPE  
 KOCTA  
 KODD  
 KOPDEF  
 KPACKEDE  
 KPAGE  
 KPRINT  
 KPSECT  
 KQUAD  
 KREF1  
 KREF16  
 KREF2  
 KREF4  
 KREF8  
 KREPT  
 KRESTORE  
 KSAVE  
 KSBTTL  
 KSGNB  
 KSGNW  
 KTITLE  
 KVECTOR  
 KWARN  
 KWEAK  
 KWORD  
 KXFER  
 LBR\$GET\_RECORD  
 LBR\$INI\_CONTROL  
 LBR\$LOOKUP\_KEY  
 LBR\$OPEN  
 LIB\$GET\_VM  
 LIBRY  
 LSTSG\_MACRODEF  
 LSTSK\_BUFSIZ  
 LSTSK\_L\_P PAGE  
 LSTSK\_TITCE SIZ  
 MAC\$ABORT\_PASS1  
 MAC\$AB\_LINEBF  
 MAC\$AB\_TMPBUF  
 MAC\$AB\_TMPSYM  
 MAC\$AL\_1 PAGE  
 MAC\$ALL\_MCB\_MLF  
 MAC\$AL\_PRMH5HTB  
 MAC\$AL\_UMCHSHTB  
 MAC\$DEX\_1 PAGE  
 MAC\$ERRRPT

= 00000085  
 = 00000061  
 = 0000003C  
 = 00000050  
 = 00000051  
 = 00000054  
 = 00000052  
 = 00000063  
 = 00000064  
 = 0000007A  
 = 00000062  
 = 00000074  
 = 00000083  
 = 0000005C  
 = 00000075  
 = 00000036  
 = 00000065  
 = 00000072  
 = 00000066  
 = 0000003D  
 = 0000006D  
 = 00000084  
 = 0000006E  
 = 0000006F  
 = 00000070  
 = 0000004D  
 = 00000067  
 = 00000068  
 = 0000006B  
 = 0000007C  
 = 0000007D  
 = 00000069  
 = 00000059  
 = 00000073  
 = 0000006C  
 = 0000003E  
 = 00000077  
 \*\*\*\*\* X 05  
 000001B8 RG X 05  
 \*\*\*\*\* X 05  
 = 00000086  
 = 0000003C  
 = 00000028  
 \*\*\*\*\* X 05  
 000002DB RG X 05  
 \*\*\*\*\* X 05  
 \*\*\*\*\* X 05  
 \*\*\*\*\* X 05  
 \*\*\*\*\* X 05

MAC\$ERR_LBOPEN			OBJ\$K_BUFSIZ	= 00000200
MAC\$ERR_NOMEM_0			OPFSM_LASTOPR	= 00002000
MAC\$GETCHR			OPFSM_OPTEXP	= 00001000
MAC\$GET_MLB_LIN	0000017F	RG 05	OPFSV_LASTOPR	= 0000000D
MAC\$GL_BASEADDR			OPFSV_OPTEXP	= 0000000C
MAC\$GL_ERRPT			PSC\$B_NAME	00000004
MAC\$GL_ERRPTX			PSC\$B_SEG	0000000C
MAC\$GL_INPUTP			PSC\$B_UNUSED	0000000B
MAC\$GL_LIBFUNC			PSC\$K_BLKSIZE	00000013
MAC\$GL_LIBTYPE			PSC\$K_NO_OPTS	= 0000000A
MAC\$GL_LINELN			PSC\$L_CURLOC	0000000F
MAC\$GL_LINENUM			PSC\$L_LINK	00000000
MAC\$GL_LINEPT			PSC\$L_MAXLGTH	00000005
MAC\$GL_MACPTR			PSC\$M_ABS	= FFFFFFF7
MAC\$GL_MLB_CNT			PSC\$M_ALIGNFLG	= 00004000
MAC\$GL_MLB_GET			PSC\$M_ALLOPTNS	= 000003FF
MAC\$GL_MLB_MDF			PSC\$M_BYT	= 00004000
MAC\$GL_MLB_QUE			PSC\$M_CON	= FFFFFFFB
MAC\$GL_MLFPTR			PSC\$M_DEFAULT	= 000001C8
MAC\$GL_TXRFA			PSC\$M_EXE	= 000000C0
MAC\$GQ_LINEBFDS			PSC\$M_GBL	= 00000010
MAC\$IMPLMCALL	00000047	RG 05	PSC\$M_LCL	= FFFFFFEF
MAC\$INTOUT_2_LW			PSC\$M_LIB	= 00000002
MAC\$LAST_CHANCE			PSC\$M_LONG	= 00004800
MAC\$MLB_DEFNAM			PSC\$M_NOEXE	= FFFFFFBF
MAC\$SKIPSP			PSC\$M_NOPIC	= FFFFFFFE
MAC\$SRCSYMTAB			PSC\$M_NORD	= FFFFFF7F
MAC\$SYMSCNUP			PSC\$M_NOSHR	= FFFFFFDF
MAC\$SYSLIB_DFN			PSC\$M_NOVEC	= FFFFFDFF
MAC\$SYSLIB_FNM			PSC\$M_NOWRT	= FFFFFEFF
MAC\$SYSLIB_MLF			PSC\$M_OVR	= 00000004
MAC\$SYSLIB_SET	0000026E	RG 05	PSC\$M_PAGE	= 00006400
MAC\$TMPSYMDS	00000000	R 04	PSC\$M_PIC	= 00000001
MAC\$CANTLOCMAC	= 007D905A		PSC\$M_QUAD	= 0004C00
MAC\$DIRSYNX	= 007D906A		PSC\$M_RD	= 00000080
MAC\$MACLBFMTER	= 007D913A		PSC\$M_REL	= 00000008
MAC\$MLBOPNERR	= 007D923A		PSC\$M_SHR	= 00000020
MAC\$UNTERMARG	= 007D922A		PSC\$M_USR	= FFFFFFFD
MAC\$R			PSC\$M_VEC	= 00000200
MAC\$TXT		X 05	PSC\$M_WORD	= 0004400
MAC_LIB_FMT_ERR	= 0000000D		PSC\$M_WRT	= 00000180
MAC_SUBSYS	00000174	R 05	PSC\$S_ALIGNMENT	= 00000004
MCALL	= 0000007D		PSC\$V_ALIGNMENT	= 0000000E
MLFSK_BLKSIZE	00000000	RG 05	PSC\$V_EXE	= 0000000A
MLFSK_RSFNLN	= 00000177		PSC\$V_GBL	= 00000006
MLFSL_CINDEX	= 000000FF		PSC\$V_LIB	= 00000001
MLFSL_MCDEF	00000014		PSC\$V_OVR	= 00000002
MLFSL_QLINK	00000008		PSC\$V_PIC	= 00000000
MLFSQ_FNAMDS	00000000		PSC\$V_RD	= 00000007
MLFST_FNAM	0000000C		PSC\$V_REL	= 00000003
MLFSX_NAMBLK	00000078		PSC\$V_SHR	= 00000005
MLF_ARGLIST	00000018		PSC\$V_VEC	= 00000009
MLE_SIZE	00000000	R 03	PSC\$V_WRT	= 00000008
NAMSC_BID	= 00000002		PSC\$W_FLAG	= 00000009
NAMSC_BLN	= 00000060		PSC\$W_OPTIONS	= 0000000D
NAMSC_MAXRSS	= 000000FF		RDX\$V_BINARY	= 00000000
NEXT_MLB	00000058	R 05		

RDX\$V\_DECIMAL = 00000002  
RDX\$V\_DOUBLE = 00000005  
RDX\$V\_FLOAT = 00000004  
RDX\$V\_GFLOAT = 00000006  
RDX\$V\_HEX = 00000003  
RDX\$V\_HFLOAT = 00000007  
RDX\$V\_OCTAL = 00000001  
REGS PC = 0000000F  
RRREG = 00000031  
SEMI = 0000003B  
STBSK\_PG\_MISS = 0000000A  
SYMSB\_NAME = 00000004  
SYMSB\_SEG = 0000000C  
SYMSB\_TOKEN = 0000000B  
SYMSK\_BLKSIZ = 0000000D  
SYMSK\_MAXLEN = 0000001F  
SYMSK\_TWOCOL = 00000010  
SYMSL\_LINK = 00000000  
SYMSL\_VAL = 00000005  
SYMSM\_ABS = 00000010  
SYMSM ASN = 00000100  
SYMSM\_CRFO = 00020000  
SYMSM\_DEBUG = 00000020  
SYMSM\_DEF = 00000001  
SYMSM\_DELMAC = 00002000  
SYMSM\_EPT = 00000200  
SYMSM\_EXTRN = 00000008  
SYMSM\_GLOBL = 00000004  
SYMSM\_LOCAL = 00000040  
SYMSM\_ODBG = 00000400  
SYMSM\_REF = 00000080  
SYMSM\_RELSECT = 00000800  
SYMSM\_SUPR = 00040000  
SYMSM\_WEAK = 00000002  
SYMSM\_XCRF = 00010000  
SYMSV\_ABS = 00000004  
SYMSV ASN = 00000008  
SYMSV\_CRFO = 0000000D  
SYMSV\_DEBUG = 00000005  
SYMSV\_DEF = 00000000  
SYMSV\_DELMAC = 00000009  
SYMSV\_EPT = 00000009  
SYMSV\_EXTRN = 00000003  
SYMSV\_GLOBL = 00000002  
SYMSV\_LOCAL = 00000006  
SYMSV\_ODBG = 0000000A  
SYMSV\_REF = 00000007  
SYMSV\_RELSECT = 0000000B  
SYMSV\_SUPR = 0000000E  
SYMSV\_WEAK = 00000001  
SYMSV\_XCRF = 0000000C  
SYMSW\_FLAG = 00000009  
TAB = 00000009  
X1 = 00000400  
X2 = 0000000F

+-----+  
! Psect synopsis !  
+-----+

## PSECT name

```
-----  
. ABS :  
. BLANK :  
$ABSS  
MAC$RO_DATA  
MAC$RW_DATA  
MAC$RO_CODE_MAC
```

	Allocation	PSECT No.	Attributes																	
000000000	( 0.)	00 ( 0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE							
000000000	( 0.)	01 ( 1.)	NOPIC	USR	CON	REL	LCL	NOSHR	EXE	RD	WRT	NOVEC	BYTE							
00000177	( 375.)	02 ( 2.)	NOPIC	USR	CON	ABS	LCL	NOSHR	EXE	RD	WRT	NOVEC	BYTE							
00000010	( 16.)	03 ( 3.)	NOPIC	USR	CON	REL	GBL	NOSHR	NOEXE	RD	NOWRT	NOVEC	LONG							
00000008	( 8.)	04 ( 4.)	NOPIC	USR	CON	REL	LCL	NOSHR	NOEXE	RD	WRT	NOVEC	LONG							
00000310	( 784.)	05 ( 5.)	NOPIC	USR	CON	REL	GBL	NOSHR	EXE	RD	NOWRT	NOVEC	LONG							

+-----+  
! Performance indicators !  
+-----+

## Phase

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.04	00:00:01.80
Command processing	103	00:00:00.39	00:00:02.31
Pass 1	287	00:00:06.03	00:00:29.58
Symbol table sort	5	00:00:00.98	00:00:02.90
Pass 2	99	00:00:01.29	00:00:05.01
Symbol table output	60	00:00:00.26	00:00:01.51
Psect synopsis output	2	00:00:00.02	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	588	00:00:09.01	00:00:43.14

The working set limit was 1350 pages.

55114 bytes (108 pages) of virtual memory were used to buffer the intermediate code.

There were 60 pages of symbol table space allocated to hold 1042 non-local and 20 local symbols.

457 source lines were read in Pass 1, producing 24 object records in Pass 2.

17 pages of virtual memory were used to define 16 macros.

+-----+  
! Macro library statistics !  
+-----+

## Macro library name

```
$255$DUA28:[MACRO.OBJ]MACRO.MLB;1
-$255$DUA28:[SYSLIB]STARLET.MLB;2
TOTALS (all libraries)
```

## Macros defined

```
11
8
19
```

1142 GETS were required to define 19 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LI\$:\$MACLIB/OBJ=OBJ\$:\$MACLIB MSRCS:\$MACLIB/UPDATE=(ENH\$:\$MACLIB)+LIB\$:\$MACRO/LIB

0226 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

